system files for installing the fully configured operating system onto the storage device; and

reinitializing the data processing system from the storage device to install the fully configured operating system.--

REMARKS

Claims 1-32 stand rejected under 35 U.S.C. §§112 and 103. Applicants add new independent claims 33 and 34. Accordingly, claims 1-34 are in the case, of which claims 1, 18, 30, 33 and 34 are independent.

Applicants note that according to paragraph 6 of the Office Action, only claims 2-32 are rejected under 35 U.S.C. §103, and specifically that independent claim 1 is not included. Applicants confirmed by telephone interview with Examiner Kriess on September 27, 1994, that this was a typographical error by virtue of the fact that each of the rejected claims 2-17 are dependent upon claim 1 and thus contain each and every limitation of claim 1 in addition to the features of the respective dependent claim.

Applicants note with appreciation Examiner's clarification in paper number 7, mailed September 27, 1994, of the specific claims rejected under §103. It is now our understanding, therefore, that Examiner's paragraph 6 rejection of the present action is intended to include all of claims 1-32; and that each of the claims are rejected under 35 U.S.C. §103 as being unpatentable over Shinjo et al., U.S. Patent No. 5,269,022 or Thorpe, U.S. Patent No. 5,276,865, and in view of Ottman et al., U.S. Patent No. 5,142,680 or Bealkowski et al., U.S. Patent No. 5,210,875.

Applicants also note the Examiner's assessment that the drawings are in acceptable form for prosecution purposes. Formal drawings will, accordingly, be submitted upon allowance of the claims.

Independent 33 is added herewith to particularly claim the features of the invention relating to the recovery diskette which includes (A) means for booting the workstation; (B) device driver means to activate the data backup system; (C) system configuration means for configuring the workstation; and (D) recovery means for loading files from the

data backup system to the workstation. Support for claim 33, and in particular for clauses A-D, is found, for example, on page 8, lines 15-16, lines 19-22, and lines 27-34.

1. §112 Rejections

Applicants respectfully disagree with the Examiner's §112 rejections of claims 1-32, which include steps of "providing," "initializing," "loading" and/or "reinitializing." The Examiner's question in paragraph 2 of the present office action is whether these steps are performed by a data processing means, some specific means, or by a human user. The question suggests that such method steps of claims 1-32 might be more definite with limitations connected with these steps, such as "by a human user," or "by a data processing means" or by some "specific means." However, Applicants suggest that such additions are narrowing limitations which unnecessarily limit the full scope of the invention entitled to the Applicants. The breadth of independent claims 1, 18 and 30, as drafted and as amended herewith, is permitted in view of the prior art. The fact that such a claim is broad does not necessarily justify a rejection on the ground that the claim is vague and indefinite. *See* MPEP §706.03(d).

In accord with the Examiner's comments, however, Applicants include herewith new independent claim 34 which is similar to amended claim 1, except that the "providing" steps are incorporated into the steps of "initializing" and "loading."

Claims 18 and 30 are amended in accordance with the Examiner's §112 rejections set forth in the fourth paragraph of the present action. More specifically, claims 18 and 30 are amended, respectively, after clause 4 (claim 18) and after clause 2 (claim 30) to fix the typographical error by replacing the "period" with a semi-colon.

In view of the foregoing remarks and amendments, Applicants respectfully request reconsideration as to the §112 rejections of paragraphs 2 - 4 of the present Office Action.

Independent claims 1, 18 and 30 are amended to recite a method for loading a "fully configured" operating system to a data processing storage device. Accordingly, by amendment, "first operating system" has been replaced by "fully configured operating system" in each of claims 1, 18 and 30. Support for such an amendment may be found, for example, in the specification on page 3, lines 5-7 and lines 17-20. Other minor

A

amendments were made in claim 1 for clarity and which either rearranged words or deleted unnecessary words. Please also note that new independent claims 33 and 34 have similar usage of a "fully configured operating system."

Likewise, independent claims 1, 18 and 30 are also amended to recite a method for loading a "temporary" operating system to a data processing storage device before the storage device may be loaded and reinitialized with the fully configured operating system. Accordingly, by amendment, "second operating system" has been replaced by "temporary operating system" in each of claims 1, 18 and 30. Support for such an amendment may be found, for example, in the specification on page 3, line 30, and on page 12, lines 6-9. Please also note that new independent claims 33 and 34 have similar usage of a "temporary operating system."

Claims 5 and 13 are amended to conform to the amendments of claim 1.

2. §103 Rejections

Each of the claims stand rejected under 35 U.S.C. §103 as being unpatentable over Shinjo et al., U.S. Patent No. 5,269,022 or Thorpe, U.S. Patent No. 5,276,865, and in view of Ottman et al., U.S. Patent No. 5,142,680 or Bealkowski et al., U.S. Patent No. 5,210,875.

Applicants contend that the invention sets forth several significant structural differences from the cited references. Applicants further contend that the combination of the cited references lacks any teaching, suggestion, or incentive to construct the claimed invention or to otherwise render the claimed invention obvious.

The invention of independent claims 1, 18, 33 and 34 relates generally to time-saving apparatus and methods for restoring a fully configured operating system to a storage device of a computer workstation in the event that the workstation's operating system becomes corrupted or otherwise inoperative, such as by a storage device crash, e.g., a system disk crash. A method of the invention may also be used to configure multiple computer systems with a target configuration, such as set forth in claim 30.

3. Shinjo et al. and Thorpe

The cited references of Shinjo et al., and Thorpe, generally concern computer back-up systems which store a copy of a computer's main memory. Applicants recognize that such prior art back-up systems exist, as stated on page 2, lines 12-14 of the specification. However, "these backup systems normally require that the disk operating system installed on the hard disk be intact and fully operational before data can be restored to the hard disk", page 2, lines 16-18. Such is the case for Shinjo et al. and Thorpe.

More specifically, Shinjo et al. describe a method of booting and subsequently rebooting a computer by avoiding multiple disk accesses during the reboot procedure (see, for example, col. 1, lines 22-34, and Fig. 2). Shinjo et al. restore the data to the primary or main memory from a back-up memory to facilitate the boot process whenever the computer determines that such a back-up memory can be restored. This determination is made via a "backup flag" (col. 3, lines 13-45). Shinjo et al. do not, however, appear to recover the operating system of the computer in the event of a disk crash without incurring the long and tedious task of rebuilding the operating system on the disk - the very problem solved by the present invention in claims 1, 18, 33 and 34. Further, Shinjo et al. do not disclose a method of configuring multiple computer systems from a target configuration, such as set forth in claim 30.

Shinjo et al. suggest that either the system memory be twice as large as the main memory (see col. 2, lines 39-42) or that a mirror image of the main memory is stored onto disk (see Fig. 3). The goal of Shinjo et al. is to maintain a mirror image of the main memory, including the resident operating system and the running environment of any application programs, as a backup. However, in the event of a computer crash which affects the resident operating system, Shinjo et al. do not disclose operative control within the computer to re-acquire and load the mirror image. Hence, Shinjo et al. do not appear to be suitable for (i) providing a fully configured operating system to a computer in the event of a disk crash affecting the operating system, as provided in claims 1, 18, 33 and 34; or (ii) configuring multiple computer systems from a target computer system, such as provided in claim 30.

Thorpe relates to a modified ON/OFF switch of a personal computer. In accord with Thorpe, when a computer operator switches the computer's power switch to OFF,

the computer remains powered until a backup memory of the computer memory is made; and thereafter a power relay is activated to disconnect the computer from power. Thorpe thus discloses a method of automatically performing a backup without operator intervention upon power-down. The reference does not, however, teach, disclose or otherwise suggest, alone or in combination with the other references cited by the Examiner, how to restore a "fully configured operating system" in the event of a crash of the storage device, e.g., disk, or in the event that the operating system becomes corrupted, such as set forth by Applicants in the invention of claims 1, 18, 33 and 34. Further, the reference does not disclose a method of configuring multiple computer systems from a target configuration, such as set forth in claim 30.

In contrast to Shinjo et al. and Thorpe, the present invention of claims 1, 18, 33 and 34 relates to apparatus and methods for restoring system and application files to the storage device, e.g., a disk drive, of a computer workstation in the event of a storage device crash, or in the event that there is corruption of those files residing on the storage device. In such an event, a "temporary operating system" is installed from a recovery diskette, i.e., the "second media," together with a number of files, i.e., the "configuration-specific data files," necessary to support transfer of data between the storage device and the back-up system or "first media," e.g., a QIC or DAT tape cartridge located in a backup device such as a tape drive. With the temporary operating system installed, the first media is accessed and an entire set of "fully configured operating system files" is transferred from the first media to the storage device. Once the storage device is properly configured, the computer workstation is rebooted, i.e., "reinitialized," from the storage device and operation to the workstation is fully restored.

Similarly, and unlike Shinjo et al. and Thorpe, the invention of claim 30 relates to a method of configuring the disk drives of multiple computer systems from a target configuration. A "temporary operating system" is installed onto each computer system from a "second media," together with a number of files, i.e., the "configuration-specific data files," necessary to support transfer of data between each computer system and the "first media" of the target configuration. With the temporary operating system installed, the first media is accessed and an entire set of "fully configured operating system files" is transferred from the first media to each computer system.

4. Ottman et al.

Ottman et al. disclose a method for using the configuration of a "transferor" personal computer on a network to configure other "transferee" personal computers on the network. In particular, a subset of network communications operating system files are loaded into a "root" directory of the transferee computer's RAM memory. The transferee computer is thereafter booted on the network - operating on its RAM root directory - and downloaded with a "new version" of the operating system on its hard disk. *See col. 2, lines 20-47.*

The invention of independent claims 1, 18, 30, 33 and 34 differs from Ottman et al., for example, in that Ottman et al. use a network link to transfer a subset of the operating system and network communication code to the transferee computer's RAM memory. Once connected via the network, the files comprising the operating system are transferred by the transferor computer to the transferee computer's disk drive. In Ottman et al., therefore, the sequence of events to load the target operating system from the transferor computer to the transferee computer is as follows: (1) the transferee computer is booted so that it is operating under a disk-based operating system; (2) the transferee computer is connected in network with the transferor computer; (3) a subset of the operating system and the necessary network communication files is installed into a root directory in the transferee computer's RAM to enable network communication (thus control is passed from the disk-based operating system to the RAM resident operating system subset); (4) the transferee computer is rebooted under network control of the root directory; and (5) a copy of the transferor operating system is transferred, via network, to the transferee computer without affecting the root directory RAM memory.

Ottman et al. do not disclose apparatus and methods for restoring the disk-based operating system in the event of a disk crash or corruption of the operating system that would otherwise render the system un-bootable, such as provided for in the invention claimed by Applicants in claims 1, 18, 33 and 34. Further, Ottman et al. do not disclose an initialization and loading procedure as claimed by Applicants in claims 1, 18, 30 and 34.

In particular, note that Applicants claims 1, 18, 30 and 34 include the steps of (1) "initializing the ... data processing system from the second media ... to provide a temporary operating system and using the configuration-specific data files to configure the

data processing system" and (2) "loading the fully configured operating system files from the first media ... using the temporary operating system" and such steps (1) and (2) are absent from Ottman et al.

Instead, the loading of the operating system into the transferee computer of Ottman et al. is accomplished by the transferor computer and not by the transferee computer. *See col. 4, lines 27-31.* Applicants specifically claim "loading the fully configured operating system files from the first media ... using the temporary operating system," claims 1, 18, 30 and 34.

Further, note that Ottman et al. lack any disclosure of a recovery diskette, as claimed by Applicants in claim 33. Independent claim 33 recites a diskette that is insertable into a workstation and that boots, activates, configures and recovers the fully configured operating system of that workstation from a backup system, e.g., a tape drive. The transferor computer of Ottman et al. is not required or appropriate in connection with claim 33.

In accord with the invention, therefore, the temporary operating system is booted via the second media, e.g., a floppy disk, and the remaining operating system folders are restored from the first media, e.g., the backup tape, by the data processing system, e.g., the computer workstation. This solves the problem of how to boot a system that has no operating system or which has a corrupted operating system. Further, the temporary operating system is configured by the data processing system using the configuration-specific data of the second media; and the fully configured operating system is loaded by the data processing system using the temporary operating system.

5. Bealkowski et al.

The reference of Bealkowski et al. concerns a BIOS load for computer systems, where BIOS specifically refers to the BASIC input/output system code for allowing new devices to be added to the computer system with relative freedom from hardware peculiarities. In particular, Bealkowski et al. disclose the modification of BIOS by loading modifiable portions of BIOS onto a direct access storage device which can be loaded into the computer system when required. *See col. 2, lines 66-68, and col. 3, line 1-11.*

In Bealkowski et al., the BIOS code is split into a first portion, in ROM, which initializes the data processor and the direct access storage device to read a master boot record from the direct access storage device into the random access memory. The first portion of BIOS confirms that the master boot record is compatible with the system hardware by verifying similarity with the data segment portion of the master boot record. If compatible, the first BIOS portion causes the system processor to execute the executable code portion of the master boot record and to load the remaining BIOS from the storage device into RAM for execution therein. Once the master boot record is resident in RAM, control is passed thereto to boot up the operating system from the disk drive, and the ROM-based BIOS is abandoned. *See col. 3, lines 22-55.*

Bealkowski et al. do not disclose, alone or in combination with the other cited references, apparatus or methods for restoring the disk-based operating system in the event of a disk crash or a corruption of the operating system that would otherwise render the system un-bootable, such as provided for in the invention claimed by Applicants in claims 1, 18, 33 and 34. Further, Bealkowski et al. do not disclose or suggest an initialization and loading procedure as claimed by Applicants in claims 1, 18, 30 and 34.

Dependent claims 2-17 incorporate each of the steps and limitations of independent claim 1, argued above, as well as the additional features set forth in such dependent claims. Therefore, Applicants believe that claims 2-17 are allowable over the cited art, without further amendment.

Likewise, dependent claims 19-29 incorporate each of the steps and limitations of independent claim 18, argued above, as well as the additional features set forth in such dependent claims. Therefore, Applicants believe that claims 19-29 are allowable over the cited art, without further amendment.

Dependent claims 31-32 incorporate each of the steps and limitations of independent claim 30, argued above, as well as the additional features set forth in such dependent claims. Therefore, Applicants believe that claims 31-32 are allowable over the cited art, without further amendment.

Each of the cited references lacks disclosure corresponding to Applicants' new independent claim 33 relating to a recovery diskette. Specifically, independent claim 33

recites a recovery diskette that is insertable into a workstation and includes (i) means for booting the workstation under a temporary operating system and (ii) device driver means to activate the data backup system. The diskette further includes (iii) configuration means for configuring the workstation and (iv) recovery means for loading files from the backup system to the workstation. In combination, these elements restore a fully configured operating system to the workstation upon reinitialization. Such a combination is particularly absent from the cited references of Shinjo et al. or Thorpe in view of Ottman et al. or Bealkowski et al.

The foregoing structural and functional/operation distinctions, and the amendments filed herewith, establish that claims 1-34 are not rendered obvious by Shinjo et al. or Thorpe in view of Ottman et al. or Bealkowski et al.

In summary, the above amendments and remarks place the application in condition for allowance with claims 1-34. Reconsideration of this application and early allowance are accordingly requested. If there are any remaining issues, an opportunity for interview is requested. We invite the Examiner to call the undersigned at the telephone number indicated below.

Respectfully submitted,

Testa, Hurwitz & Thibeault

Curtis A. Vock

Curtis A. Vock
Reg. No. 38,356
Attorney for Applicants

Dated: October 28, 1994

Exchange Place
53 State Street
Boston, MA 02109-2809
(617) 248-7367